

Absolvování individuální odborné praxe

Individual Professional Practice in the Company

Zadání bakalářské práce

Student: **Lukáš Kielar**

Studijní program: B2647 Informační a komunikační technologie

Studijní obor: 2612R025 Informatika a výpočetní technika

Téma: Absolvování individuální odborné praxe
Individual Professional Practice in the Company

Zásady pro vypracování:

1. Student vykoná individuální praxi ve firmě: BINARGON s.r.o.
2. Struktura závěrečné zprávy:
 - a) Popis odborného zaměření firmy, u které student vykonal odbornou praxi a popis pracovního zařazení studenta.
 - b) Seznam úkolů zadaných studentovi v průběhu odborné praxe s vyjádřením jejich časové náročnosti.
 - c) Zvolený postup řešení zadaných úkolů.
 - d) Teoretické a praktické znalosti a dovednosti získané v průběhu studia uplatněné studentem v průběhu odborné praxe.
 - e) Znalosti či dovednosti scházející studentovi v průběhu odborné praxe.
 - f) Dosažené výsledky v průběhu odborné praxe a její celkové zhodnocení.

Seznam doporučené odborné literatury:

Podle pokynů konzultanta, který vede odbornou praxi studenta.

Formální náležitosti a rozsah bakalářské práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí bakalářské práce: **Ing. Jan Gaura**

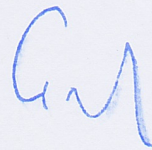
Konzultant bakalářské práce: Mgr. Jiří Svoboda

Datum zadání: 01.09.2013

Datum odevzdání: 07.05.2014



doc. Dr. Ing. Eduard Sojka
vedoucí katedry



prof. RNDr. Václav Snášel, CSc.
děkan fakulty

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

V Ostravě 7. května 2014

Kiehn

Tímto chci poděkovat Mgr. Jiřímu Svobodovi za poskytnutí příležitosti vykonávat odbornou praxi ve firmě BINARGON s.r.o. a za vedení v průběhu této praxe. Také bych rád poděkoval Ing. Janu Gaurovi za konzultace a rady důležité pro tvorbu závěrečné zprávy.

Abstrakt

Bakalářská práce pojednává o průběhu odborné praxe ve firmě BINARGON s.r.o. Nejprve jsou zde rozebrány obory činnosti této společnosti a softwarové nástroje používané v průběhu praxe. Následuje základní seznámení s využitými technologiemi (především programovacími jazyky) a jejich konkrétním užíváním při vypracovávání jednotlivých zadání. Poté jsou podrobněji popsány úkoly, na kterých jsem pracoval, a to konkrétně tvorba šablony pro web internetového obchodu a vytváření modulů do administrace internetového obchodu k napojení na rozhraní poskytovaná společností Google. Posledním zadáním v rámci této praxe je tvorba modulů, které by dohromady byly schopny zastoupit specializované programy pro účetnictví.

Klíčová slova: Internetový obchod, XSL transformace, XML, Webmaster Tools, Google Analytics, Mzdové účetnictví, HTML5

Abstract

The bachelor's thesis discusses the course of professional practice in company called BINARGON s.r.o. Firstly, there is an analysis of fields of activity of the company and software products used during the course of the practice. Following is a basic introduction to used technologies (mainly programming languages) and their specific usage in development of particular objectives. Then there are further details of tasks I was assigned to, which were namely a production of a template for e-shop and the creation of modules for the administration of e-shop intended to connect it to interfaces provided by Google. The last assignment in the course of this practice is the production of modules that would be able to replace specialized accounting software.

Keywords: E-shop, XSL Transformations, XML, Webmaster Tools, Google Analytics, Payroll Accounting, HTML5

Seznam použitých zkratek a symbolů

API	– Application Programming Interface
CSS	– Cascading Style Sheets
GA	– Google Analytics
HTML	– Hypertext Markup Language
HTTP	– Hypertext Transfer Protocol
JSON	– JavaScript Object Notation
PHP	– Hypertext Preprocessor
SEO	– Search Engine Optimization
SGML	– Standard Generalized Markup Language
SQL	– Structured Query Language
WT	– Webmaster Tools
XML	– Extensible Markup Language
XSLT	– Extensible Stylesheet Language Transformation

Obsah

1	Úvod	4
2	O firmě	5
3	Použité technologie a vývojová prostředí	6
3.1	PHP	6
3.2	HTML a CSS	7
3.3	XSLT	8
4	Vypracované úkoly	10
4.1	Tvorba šablony	10
4.2	Webmaster Tools	11
4.3	Google Analytics	13
4.4	Personalistika a mzdové účetnictví	15
5	Užití dovedností v praxi	20
6	Chybějící dovednosti	21
7	Shrnutí výsledků	22
8	Závěr	23
9	Reference	24
	Přílohy	24
A	Šablona e-shopu	25
B	Webmaster Tools	26
C	Google Analytics	27
D	Mzdové účetnictví	29

Seznam obrázků

1	Grafické zobrazení seznamu zaměstnanců	9
2	Šablona e-shopu proti-ohni.cz – úvodní strana	25
3	Webmaster Tools - Detail webu	26
4	Google Analytics – Prohlížeče	27
5	Google Analytics – Akvizice	27
6	Google Analytics – Místa	28
7	PDF s výplatními páskami	29

Seznam výpisů zdrojového kódu

1	XML seznam zaměstnanců	8
2	HTML a XSL seznam zaměstnanců	8
3	Zpracovaný HTML seznam zaměstnanců	9
4	Výpočet zálohy na daň	17

1 Úvod

Tato práce pojednává o průběhu odborné praxe, kterou jsem absolvoval v průběhu akademického roku 2013/2014, ve firmě BINARGON s.r.o. Práce je rozdělena na kapitoly, které se zabývají oblastí působení firmy BINARGON v rámci nabídky informačních systémů, popisem jednotlivých použitých technologií a vypracovaných úkolů.

Nejprve jsou charakterizovány jazyky PHP, HTML a CSS zároveň s XSL transformacemi. Tyto jazyky, spolu s SQL, byly velmi hojně využívány v realizaci každého jednotlivého zadání. Poté je soustředěna pozornost na postupy využívané při tvorbě šablony internetového obchodu, vysvětlení používaných měřítek u vývoje zásuvných modulů pro Webmaster Tools a Google Analytics, poskytované společností Google, a v neposlední řadě je zde popsána část vývoje systému pro účetnictví.

V závěrečné části zprávy je rozebráno, jaké dovednosti nabyté během studia jsem nejvíce zužitkoval, co jsem před praxí neuměl a musel jsem se to doučit nebo danou problematiku konzultovat s odborníky a shrnutí výsledků úkolů vypracovávaných během praxe.

2 O firmě

Firma BINARGON s.r.o. nabízí a poskytuje komplexní řešení v oblasti tvorby webů a e-shopů, poskytování webhostingu a správy domén zároveň s technickou podporou a odborným školením k daným produktům a službám[1].

Každodenní náplní práce zaměstnanců je, kromě výše zmíněné technické podpory a s tím souvisejícím odstraňováním chyb nahlášených klienty, tvorba šablon pro elektronické obchody na základě grafické předlohy dodané externím profesionálním grafikem a jejich následné testování, import dat z různých formátů poskytovaných například dodavateli jednotlivých elektronických obchodů do odpovídajících databází, tvorba přídatných modulů, o které mají zákazníci zájem, případně které by mohly být novým a stávajícím klientům nabízeny.

Vzhledem k tomu, že praxe probíhala jen dva dny v týdnu, nebyly mi přiřazovány úkoly, jejichž řešení spěchalo, nýbrž spíše taková zadání, jejichž realizace mohla být roztažena na delší časové období. Také jsem nebyl součástí technické podpory, jelikož pochopit celou funkčnost systému a udržovat si povědomí o změnách by při šestnácti hodinách týdně bylo ještě mnohem náročnější než při běžné pracovní době.

3 Použité technologie a vývojová prostředí

Během své praxe jsem si značně rozšířil své vědomosti o některých technologiích a naučil jsem se pracovat s technologiemi či knihovnami, které jsem dříve neznal.

Základem mé praxe bylo programování v jazyce PHP, k čemuž jsem využil Visual Studio 2010 s rozšířením PHP Tools for Visual Studio. Data získaná pomocí PHP z vnějších zdrojů (např. API, MySQL databáze), jsem zpracoval a naformátoval do formátu XML, ze kterého jsem tvořil výstup v programu PSPad s použitím technologií HTML, XSLT a CSS. K rozdělení grafické předlohy na jednotlivé části použitelné při tvorbě šablony mi pomohl program Adobe Photoshop.

3.1 PHP

PHP je skriptovacím programovacím jazykem vyvíjeným od roku 1995. V průběhu praxe byla používána verze 5.5. Díky své jednoduchosti a zároveň širokým možnostem je nej-používanějším jazykem pro tvorbu webových aplikací[2]. Základními vlastnostmi jsou podpora jak procedurálního, tak objektově-orientovaného programování, dynamická typová kontrola a multiplatformnost.

Konkrétním použitím PHP při mé činnosti v rámci odborné praxe byla manipulace s daty především v těchto případech:

- Práce s daty v MySQL databázi (získávání, ukládání, editace a mazání dat)
- Práce s API (získávání a zasílání dat)
- Tvorba XML dokumentu, který byl zdrojem dat pro výstupní stránku

3.1.1 Třída mPDF

mPDF[3] je třídou napsanou v PHP, která umožňuje vygenerovat PDF dokument na základě HTML a CSS kódů. Pomocí této třídy jsem byl schopen data načtená například z databáze naformátovat do podoby dokumentu, který je možno vytisknout a založit pro archivní účely či odevzdat na úřadě, který daný dokument vyžaduje v konkrétním formátu. Jelikož se formulář tiskne již vyplněný a nevyplňuje se ručně, je i vhodnější z hlediska strojového čtení.

3.2 HTML a CSS

HTML spolu s CSS jsou jazyky, pomocí kterých lze rozvrhnout grafickou podobu webu.

3.2.1 HTML

HTML je značkovacím jazykem, který původně vznikl v roce 1990 jako jednodušší forma SGML. Již od roku 1999 je až do dnešní doby aktuální verze verze 4.01, nicméně už během roku 2014 by měla být oficiálně nahrazena verzí 5.

Značky HTML se dělí do tří skupin:

- Strukturální
- Sémantické
- Stylistické

Pomocí strukturálních značek můžeme rozdělit dokument na nadpisy, odstavce, odkazy, vkládat obrázky, tvořit tabulky či seznamy a podobně.

Sémantické značky dělí dokument na logické celky, kterými mohou být například záhlaví, navigace nebo obsah. Na používání těchto značek je kladen vysoký důraz právě v HTML5, jelikož mohou pomoci vyhledávačům nalézt na stránce relevantní obsah.

Stylistické značky mají za úkol upravovat vzhled prvků stránky, avšak pro tyto účely je preferováno použití kaskádových stylů, proto specifikace HTML5 již některé prvky, jako font (nastavení vzhledu písma), strike (přeškrtnuté písmo) či big (zvětšení písma), neobsahuje.

3.2.2 CSS

CSS je jazykem, který vznikl ve snaze oddělit vzhled stránky od její struktury a obsahu. Pomocí široké nabídky selektorů si lze přesně zvolit, které prvky chceme jakým způsobem nastavit, ale na rozdíl od stylování pomocí HTML obvykle daný styl přiřadíme kolekci prvků a nemusíme tedy stylovat každý prvek zvlášť, což nejen zkracuje celkovou délku kódu, ale také usnadňuje případné budoucí úpravy.

Zásadním problémem při stylování je různý výklad a podpora CSS atributů různými prohlížeči. Abychom si byli jisti, že se stránka zobrazuje správným a stejným způsobem ve všech prohlížečích, musíme ji na daných prohlížečích testovat a případně upravovat, aby bylo zobrazení všude vyhovující, což mnohdy může značně zpomalovat postup například při tvorbě šablony.

3.3 XSLT

Transformace XSL umožňují na základě struktury a obsahu zdrojového XML dokumentu naformátovat a vyplnit daty strukturu stvořenou například pomocí HTML a CSS. Ačkoliv jazyk XSL obsahuje jen několik značek, dokáže plně nahradit kupříkladu cykly (například prvek `for-each` - „Pro každý XML prvek, jehož jméno je ... a nachází se ..., vypiš ...“) a podmínky (například prvek `choose` - „Jestliže atribut prvku ... obsahuje ..., vypiš ..., v opačném případě vypiš ...“), které bych jinak musel řešit třeba použitím PHP mnohem méně efektivním způsobem. K navigaci v XML je využíván jazyk XPath.

Konkrétním příkladem použití XSLT může být výpis seznamu zaměstnanců. Konkrétní část zdrojového XML vytvořená PHP skriptem bude vypadat kupříkladu takto:

```
<employees>
  <employee id="2" name="Karel" surname="Capek" state="current"/>
  <employee id="10" name="Lukas" surname="Kielar" state="current"/>
  <employee id="11" name="Jana" surname="Sekaninova" state="current"/>
  <employee id="3" name="Herbert.George" surname="Wells" state="current"/>
  <employee id="1" name="Vaclav" surname="Zeleny" state="past"/>
</employees>
```

Výpis 1: XML seznam zaměstnanců

XSL kód, který pracuje se zdrojovým XML dokumentem:

```
<xsl:template name="list">
  <div class="employees_menu">
    <a href="index.php?funkce=99">
      <h2>Seznam zamestnancu</h2>
    </a>
    <xsl:for-each select="/gsdata/employees/employee">
      <xsl:if test="@state='_current'">
        <a href="index.php?funkce=99&detail={@id}&employees={$employees}">
          <xsl:choose>
            <xsl:when test="@id='_1'">
              <xsl:attribute name="class">active</xsl:attribute>
            </xsl:when>
            </xsl:choose>
            <xsl:value-of select="@surname" />, <xsl:value-of select="@name" />
          </a>
        </xsl:if>
      </xsl:for-each>
      <xsl:choose>
        <xsl:when test="$employees='_past'">
          <a href="index.php?employees=current">Soucasni zamestnanci</a>
        </xsl:when>
        <xsl:otherwise>
          <a href="index.php?employees=past">Archiv zamestnancu</a>
        </xsl:otherwise>
      </xsl:choose>
    </div>
  </xsl:template>
```

Výpis 2: HTML a XSL seznam zaměstnanců

Výsledný HTML dokument:

```
<div class="employees_menu">
  <a href="index.php?funkce=99">
    <h2>Seznam zamestnancu</h2>
  </a>
  <a href="index.php?funkce=99&detail=2&employees=current">Capek, Karel</a>
  <a href="index.php?funkce=99&detail=10&employees=current">Kielar, Lukas</a>
  <a href="index.php?funkce=99&detail=11&employees=current">Sekaninova, Jana</a>
  <a href="index.php?funkce=99&detail=3&employees=current">Wells, Herbert George
  </a>
  <a href="index.php?employees=past">Archiv zamestnancu</a>
</div>
```

Výpis 3: Zpracovaný HTML seznam zaměstnanců



Obrázek 1: Grafické zobrazení seznamu zaměstnanců

4 Vypracované úkoly

4.1 Tvorba šablony

Mým prvním úkolem v průběhu odborné praxe bylo vyzkoušet si vytvořit šablonu pro e-shop (náhled v příloze A). V rámci tohoto úkolu jsem se především učil pracovat s technologií XSLT, kterou jsem dříve nepoužíval, ale také jsem se zdokonaloval v práci s HTML a CSS.

4.1.1 Grafická podoba webu

Vizuální část šablony se zpracovává podle PSD souboru dodávaného grafikem ke konkrétnímu e-shopu, ze kterého lze získat informace o rozměrech a barvách jednotlivých částí grafické podoby, stylech a velikostech písem (fontů) použitých na různých částech webu a v neposlední řadě z něj lze také „vystříhat“ obrázky, které nelze nahradit pomocí kaskádových stylů. Při tvorbě vzhledu je kladen velký důraz na minimalizaci počtu obrázků na stránce a jejich suplování jinými HTML a CSS prvky, jelikož soubory obrázků jsou řádově mnohem objemnější z hlediska velikosti, než čistý text, tudíž zpomalují načítání stránky, což by potencionální zákazníci – především ty, kteří přicházejí na web z mobilních zařízení - mohlo od nákupu lehce odradit.

4.1.2 Součásti šablony

Šablona webu se skládá z několika částí (ve firmě označovaných jako funkce), které reprezentují jednotlivá zobrazení. V případě e-shopu mohou těmito částmi být například hlavní stránka, detail produktu, registrace zákazníka či jednotlivé kroky vedoucí k potvrzení objednávky produktů, které jsou obsahem košíku. Tyto části obvykle mívají společné záhlaví, postranní menu i zápatí a mění se jen obsahová část, což konkrétně znamenalo, že jsem vytvořil samostatné XSLT prvky template pro tyto neměnné části a template prvky reprezentující jednotlivé „funkce“ jsem poté jen vkládal pomocí prvku call-template mezi ně podle toho, kde se v rámci webu uživatel v danou chvíli nacházel.

4.1.3 Uspořádání kódu

Z hlediska SEO, tedy optimalizace webů pro vyhledávače, je vhodné psát šablonu tak, aby vyhledávač při jejím čtení našel podstatné informace co nejrychleji, aby se nestávalo, že je ve výsledku nakonec vůbec nenajde a zákazníkovi tedy na zadaný dotaz nebude schopen konkrétní web nabídnout. Z tohoto důvodu se postupovalo takovým způsobem, že ve výsledném kódu byl nejprve vypsán obsah a například hlavička stránky nebo horní menu, které obvykle jsou spíše grafického, respektive informačního charakteru, se v kódu nacházely až za obsahem. Vyhledávač tedy při čtení stránky narazil nejprve na produkty či popis nabídky daného obchodu, což jsou při vyhledávání relevantní informace.

V dnešní době lze stále používat výše zmíněný přístup, avšak rozšiřování HTML5 nám přináší sémantické tagy, jakými jsou například header pro záhlaví, nav pro menu (navigaci) nebo article pro obsah (článek). Web tedy můžeme psát logicky shora dolů a vyhledávač by měl být schopen najít si v kódu dané značky a indexovat důležité informace.

4.1.4 Texty

Veškeré statické texty na stránce, jakými jsou například popisy tlačítek, nadpisy či položky menu, se ukládají jako proměnné do samostatného souboru, ze kterého jsou poté pouze vkládány. Tato skutečnost usnadňuje tvorbu více jazykových verzí daného webu.

4.2 Webmaster Tools

Mým druhým úkolem bylo napojit administraci elektronického obchodu na službu Webmaster Tools, kterou poskytuje společnost Google. Tato služba využívá robota, který prochází weby registrovaných uživatelů s využitím jimi poskytnutých „map“ i vlastní umělé inteligence. Data, která jsou získána procházením daných webů, jsou především využita ve vyhledávači Google, ale prostřednictvím jejich webového rozhraní jsou také uživatelům zobrazována. Tyto informace jsou především důležité pro webové vývojáře, jelikož se podle nich může pokusit přizpůsobovat daný web tak, aby například odstranil nalezené chyby, nebo pozměnil nalezená klíčová slova tak, aby byla relevantní pro vyhledávání potenciálním zákazníkem. Pro majitele/obsahu e-shopu je také výhodné tyto informace sledovat, protože ačkoliv nemá přístup přímo ke zdrojovému kódu svého webu, může v případě jakýchkoliv nesrovnalostí kontaktovat technickou podporu, aby vyšetřila příčinu daného problému.

Google má samozřejmě vlastní webové rozhraní, kde lze výše zmíněná data prohlížet, nicméně není to zrovna „po ruce“ a pracovníci e-shopu by museli myslet na to, že je potřeba se tam čas od času podívat. Napojení na Webmaster Tools bylo realizováno v novém modulu (či pluginu, doplňku) pro administrační část elektronického obchodu za použití Google Webmaster Tools API[4], tedy rozhraní, které Google poskytuje, aby bylo možné získat jím shromážděná a naformátovaná data, poté je zpracovat a zobrazit. Náhled detailu webu v tomto modulu je obsažen v příloze B.

4.2.1 OAuth 2.0

Aby k datům, která Google poskytuje přes API, mohli přistupovat pouze oprávnění uživatelé, je potřeba se při přístupu k nim dostat skrze autorizační proces, který je zprostředkováván protokolem OAuth, konkrétně verze 2. Komunikace během autorizace je uskutečňována použitím HTTP požadavků a odpovědí. Nejprve aplikace, která žádá o přístup k datům, zašle své identifikační údaje, kterými jsou takzvané Client ID a Client Secret. Server si ověří správnost údajů a odpoví autorizačním kódem. Klient poté může autorizační kód znovu poslat serveru a tím jej „vyměnit“ za přístupový token. Spolu s přístupovým tokenem mu server také poskytne token pro obnovu přístupového

tokenu, jelikož přístupový token má omezenou trvanlivost na 3600 sekund (1 hodinu) a bez tokenu pro obnovu by bylo po této době nutné vždy projít celým autorizačním procesem znovu. Pomocí přístupového tokenu se poté lze dotazovat serveru na konkrétní informace, které v dané chvíli požadujeme.

4.2.2 Poskytované údaje

Komunikace se serverem probíhá, stejně jako v případě OAuth, prostřednictvím HTTP požadavků (konkrétně GET, POST, PUT a DELETE) a odesílaná i přijímaná data jsou ve formátu XML. Níže popíšu všechny možnosti Webmaster Tools API, kterých jsem při tvorbě modulu využil.

XML kód, který rozhraní Google vrací, obsahuje řadu jmenných prostorů (namespaces), jakými jsou například wt, atom či openSearch. Při připojení tohoto XML ke zbytku XML, podle kterého se měl generovat výstup, se však ukázalo, že XSLT neumí s těmito jmennými prostory zacházet, proto bylo nutné pomocí třídy DOMDocument z původního XML získat data a vytvořit z nich vlastní XML, se kterým již XSLT neměly problémy.

4.2.2.1 Seznam webů (Sites) Je základním kamenem práce s Webmaster Tools API, jelikož obsahuje, kromě jiných informací, ID webů, které se na daném účtu nacházejí. Pomocí nich je poté možné získávat informace, které se vztahují konkrétně k daným webům.

Další důležitou informací obsaženou v XML seznamu webů je stav verifikace daného webu vzhledem k Webmaster Tools účtu. Web musí být verifikován, aby bylo zřejmé, že majitel WT účtu je zároveň majitelem webu. Existují dva způsoby verifikace – použitím HTML stránky, kterou je nutno vložit do hlavního adresáře webu, a metatagem vloženým do hlavičky hlavní stránky webu. Ověření lze poté vyvolat zasláním PUT požadavku, který obsahuje především způsob ověření „htmlpage“, nebo „metatag“.

Weby lze samozřejmě k WT účtu pomocí API také přidávat prostřednictvím požadavku POST nebo je mazat požadavkem DELETE.

4.2.2.2 Zprávy (Messages) Vztahují se k celému účtu Webmaster Tools a informují například o dlouhodobě přetrvávajících chybách. V případě testovacího webu, který jsem při tvorbě modulu používal, opakovaně přicházely zprávy s předmětem „Zvýšený počet chyb Nenalezeno“, které se generovaly na základě Chyb při procházení.

Kromě načtení zpráv pomocí požadavku GET lze také označit vybranou zprávu jako přečtenou, nebo nepřečtenou zasláním krátkého XML kódu obsahujícího identifikační adresu zprávy jako požadavek PUT. Vybranou zprávu je možné také smazat za použití požadavku DELETE.

4.2.2.3 Klíčová slova (Keywords) Robot, který prochází web, zaznamenává nalezená slova a dělí je do dvou seznamů – na interní a externí. Interní klíčová slova jsou ta, která se nalézají přímo na webu a v případě elektronických obchodů by jimi ideálně měly být názvy kategorií a produktů. Externí klíčová slova se na stránku dostanou skrze reklamu a jiné objekty, na které web pouze odkazuje.

Během testování jsem zjistil, že interní klíčová slova přicházejí z API bez posledního písmene, nicméně nepodařilo se mi zjistit, co je toho příčinou.

4.2.2.4 Mapy webů (Sitemaps) Mapa webu je seznamem stránek na webu, který může využívat robot, aby s jistotou prošel všechny stránky, u kterých chceme, aby je prohledal. Mapa je soubor ve formátu XML, na který WT pouze poskytneme odkaz pomocí požadavku POST. Poskytnutá mapa by se měla objevit v seznamu map webů, který lze získat požadavkem GET, avšak i v tomto případě (podobně jako u interních klíčových slov) se zde vyskytla chyba, která se projevovala tak, že mapa, která byla k testovacímu webu přidána skrze originální webové rozhraní dávno předtím, než byl zahájen vývoj propojovacího modulu, se v seznamu map neobjevila. Nově přidané mapy však v seznamu map obsaženy byly a bylo je i možné vymazat užitím požadavku DELETE.

4.2.2.5 Chyby při procházení (Crawl Issues) Když robot prochází web, zaznamenává nalezené chyby. Seznam těchto chyb lze získat zasláním požadavku GET. V případě testovacího webu byly nejčastěji nalezeny chyby s kódem 404 – tedy „Stránka nenalezena“. K těmto chybám jsou poskytovány informace o tom, kdy a na jaké stránce byla chyba nalezena a odkud na tuto stránku bylo odkazováno. S těmito informacemi tedy vývojář daného webu může závadu velmi jednoduše odstranit.

4.3 Google Analytics

Následující úkol spočíval v napojení administrace elektronického obchodu na další službu poskytovanou společností Google - konkrétně Google Analytics. Skrze tuto službu mohou majitelé webů sledovat statistické informace o svých stránkách.

Propojení bylo opět realizováno v samostatném modulu pro administraci e-shopu za použití Google Analytics API[5].

4.3.1 Poskytované údaje

Ověření identity uživatele, který má v úmyslu k datům prostřednictvím API přistupovat, stejně jako v případě Webmaster Tools probíhá pomocí OAuth 2.0 (viz kapitola 4.2.1). Komunikace s rozhraním také využívá HTTP požadavků, avšak v tomto případě jsou data vrácena ve formátu JSON. V PHP lze tento formát jednoduše zpracovat pomocí funkce `json_decode`, která z původního zápisu JavaScriptového objektu vytvoří obyčejné pole, jímž lze poté pohodlně procházet a získat tak potřebná data.

Množství údajů, které lze z Google Analytics získat, jakož i způsoby jejich třídění jsou značně rozsáhlé, tudíž jsem využil jen některé z nich (výsledné grafické zobrazení

v příloze C). Filtry jsou děleny na dvě skupiny, z nichž jednu Google nazývá „metrikami“ a druhou „dimenzemi“. První skupina obvykle určuje, jaké statistické údaje chceme získat, zatímco druhá kupříkladu určuje, po jakých časových úsecích by tyto údaje měly být rozděleny. Kromě dimenzí omezuje snad kteroukoliv vybranou metriku počáteční a koncové datum – tedy ohraničení časového úseku, za který chceme získat statistiku – a maximální počet navracených záznamů – to může v případě rozumného užívání zajistit, že se v případě zvolení dlouhého časového období prohlížeč nepřetíží.

4.3.1.1 Návštěvnost První kategorii jsem pojmenoval Návštěvnost. Je založena na metrice visits a znázorňuje počet návštěv daného webu. Logickým výběrem dimenzí byly časové úseky, po kterých byly počty návštěv vráceny – defaultně je zobrazena statistika za období předcházejících dvou měsíců po jednotlivých dnech, avšak je možné zvolit zobrazení po týdnech, měsících či letech nebo průřez návštěvností v průběhu dne po jednotlivých hodinách, případně průřez rokem po týdnech.

4.3.1.2 Návštěvníci Druhá zpracovávaná metrika se v angličtině nazývá visitors. Tato metrika pomocí dimenze visitorType rozlišuje návštěvníky webu na nové a vracející se.

4.3.1.3 Prohlížeče Třetí kategorie dělí návštěvníky podle dimenzí browser a browserVersion. Na základě těchto statistik se může majitel webu snadno rozhodnout, které prohlížeče a konkrétněji i jejich verze by měl jeho web zpětně podporovat, (aby třeba obchod nepřicházel o zákazníky), ale aby mohly být využity i nové technologie, které staré prohlížeče neumí využít, ale je velmi efektivní je využívat (například prvky uvedené v HTML5).

4.3.1.4 Akvizice Tato sekce poskytuje informace o tom, odkud návštěvníci webu přišli s rozdělením podle dimenzí ga:medium, ga:source, ga:sourceMedium, ga:socialNetwork, ga:keyword a ga:fullReferrer – tedy například jestli se na stránku dostali prostřednictvím vyhledávače, reklamy, či přímo, zda přišli z některé ze sociálních sítí a případně ze které, nebo pomocí jakých klíčových slov stránku vyhledali. Tato data jsou zde propojena s metrikami ga:visits, tedy kolik návštěvníků z daného zdroje přišlo, ga:transactions, tj. kolik návštěvníků z daného zdroje dokončilo objednávku v e-shopu, a ga:totalValue – jaká celková tržba se vztahuje k danému zdroji a počtu objednávek. Tento souhrn informací může správě webu naznačit, na jaký typ reklamy by se měla zaměřit a zda se jí již používané způsoby vyplácejí.

4.3.1.5 Vstupní stránky Ukazují, na které konkrétní stránky v rámci webu nejčastěji potencionální zákazníci přicházejí (dimenze ga:landingPagePath s metrikou ga:entrances). Zároveň jsou zde i uvedeny informace o tzv. míře okamžitého opuštění pomocí metriky ga:entranceBounceRate – tedy zda návštěvník opustil web okamžitě ze vstupní stránky, nebo pokračoval na další stránky v rámci vašeho webu. Díky těmto informacím je možné kupříkladu uzpůsobit sortiment tak, aby se v něm více objevovaly produkty odpovídající poptávce založené na těchto datech.

4.3.1.6 Místa Poslední zpracovaná kategorie dělí návštěvníky podle místa ve světě, odkud se na web připojovali. Data lze rozdělit užitím různých dimenzí například podle kontinentů, států či obcí. Tyto údaje mohou ukázat, zda by se vyplatilo investovat do dalších jazykových verzí webu a rozšířit tak počty návštěvníků webu a potenciálních zákazníků.

4.3.2 Google Charts

Data získaná z Google Analytics API byla zpracována na grafické uživatelské rozhraní ve formě tabulek, nicméně přehlednějším formátem výstupu jsou diagramy. Pro tento účel jsem využil Google Charts API[6], které umožňuje pomocí jednoduchého skriptu vygenerovat z JavaScriptového pole hodnot přehledný a nastavitelný graf.

Pro většinu statistik jsem využil spojnicový či sloupcový graf a pro místa jsem zvolil koláčový graf. V přehledu používaných verzí prohlížečů jsem ovšem chtěl zakomponovat graf přímo do tabulky se statistikami. Ačkoliv v originálním uživatelském rozhraní Google Analytics jsou tabulky obsahující grafy, tudíž je pravděpodobné, že je možné pomocí některého z jejich API něco podobného vytvořit, nepodařilo se mi najít, jakým způsobem to lze provést. Využil jsem tedy kombinace HTML, XSL a CSS pro vytvoření prvků div s délkou proměnné hodnoty podle počtu návštěv z dané verze webového prohlížeče.

4.4 Personalistika a mzdové účetnictví

Posledním a zároveň největším a nejobtížnějším úkolem průběhu mé odborné bakalářské praxe je vytvořit systém, pomocí kterého by si osoby pověřené účetnictvím společností vlastních elektronické obchody vytvořené firmou, ve které pracují, mohly vést účetnictví. V současné době je velká část těchto klientů závislá na ekonomickém softwaru POHODA, do něž lze vyexportovat jistá data obsažená v e-shopu, nicméně znamená to, že jsou nuceni platit jak za provoz webu, tak za program, ve kterém spravují účetnictví. Z toho důvodu bylo rozhodnuto, že by bylo vhodné vytvořit zásuvné moduly do administrace internetového obchodu, které by umožňovaly odstranit potřebu dalšího externího softwaru.

4.4.1 Personalistika

Základem je seznam zaměstnanců (zmiňovaný v kapitole 5), jenž odkazuje na karty jednotlivých zaměstnanců. Seznam je rozdělen na zaměstnance, kteří aktuálně ve firmě pracují, a ty, kteří již ze společnosti odešli.

Součástí karty (či profilu) jsou v první řadě osobní údaje zaměstnance, avšak nejdůležitější součástí z pohledu dalšího vývoje jsou informace o uzavřené pracovní smlouvě – tedy například druh a rozměry úvazku, výše mzdy, prémie a osobního ohodnocení, vyplácení záloh, zbylé dny dovolené, apod. Další důležitou součástí jsou i data týkající se zdravotního a životního pojištění, jakož i důchodového spoření. Tato data jsou poté základem pro výpočet mzdy.

Karta zaměstnance byla obohacena i o seznam daňových slev a zvýhodnění, tabulku zaznamenávající nepřítomnost zaměstnance na směně s příslušnými důvody nepřítomnosti a formuláře pro zadání příplatku za přesčasy nebo práci ve svátky a počtu odpracovaných hodin za vybraný měsíc.

4.4.2 Mzdy

Celá sekce mzdového účetnictví je založena na tom, že jsou „posbírány“ údaje o konkrétním zaměstnanci a podle nich jsou vypočítány hodnoty, z nichž je především možno vygenerovat výplatní pásku nebo výplatní list, avšak jsou využívány i při tvorbě dalších dokumentů, které jsou povinnými výstupy mzdového účetnictví, jelikož mají být uchovávány v archivu společnosti pro případ nutnosti doložení nějakých konkrétních dat po dobu třiceti let[7].

Uložení informací o mzdě za vybraný měsíc probíhá následovně:

1. Zjištění, zda mzdy za daný měsíc již nejsou uloženy v databázi – jestliže jsou, proces ukládání neproběhne.
2. Načtení základních údajů z karty zaměstnance.
3. Výpočet počtu pracovních dnů a svátků v daném měsíci – tyto hodnoty se především využívají v případě, že není zadán počet odpracovaných hodin za vybraný měsíc.
4. Zjištění průměrné hodnoty hodinové mzdy za předchozí kalendářní čtvrtletí výpočtem podílu součtu hrubé mzdy za tři měsíce předchozího čtvrtletí a součtu odpracovaných hodin za tyto měsíce. Jestliže data nejsou k dispozici (například z toho důvodu, že daný zaměstnanec neodpracoval potřebné měsíce), je průměrná hodinová mzda vypočítána z aktuální základní mzdy (v případě měsíční mzdy), resp. je přímo použita hodnota aktuální hodinové mzdy.

$$\text{Průměrná hodinová mzda} = \frac{\text{Základní měsíční mzda}}{\text{Počet hodin denně} \cdot \text{Počet dní v týdnu} \cdot 4,348} \quad (1)$$

Konstanta 4,348 vyjadřuje průměrný počet týdnů v měsíci.

5. Načtení dat o nepřítomnosti zaměstnance v průběhu daného měsíce. Tyto informace jsou opět využívány především v případě, kdy není zadán počet odpracovaných hodin, a to konkrétně tak, že jsou rozděleny do jednotlivých proměnných podle typu nepřítomnosti na nemoc, ošetrovné, neplacené volno, placené volno, dovolenou a neomluvenou absenci. Zároveň je počet dnů a hodin nepřítomnosti odečten od počtu odpracovaných dnů, resp. hodin s tím, že poté jsou znovu přičteny hodiny, které byly odpracovány v první a poslední den nepřítomnosti.

V případě nepřítomnosti z důvodu nemoci jsou zároveň přičítány náhrady, které jsou poskytovány za čtvrtý až čtrnáctý den nemoci, a to ve výši 60% z hodnoty průměrné hodinové mzdy.

6. Získání daňových slev a zvýhodnění uplatnitelných pro daného zaměstnance. Tyto položky se dělí na měsíční a roční podle toho, zda se odečítají z daní každý měsíc, nebo se zaúčtují pouze při ročním zúčtování.
7. Načtení srážek, což jsou částky, které se zaměstnanci mohou strhávat ze mzdy v průběhu několika měsíců například z důvodu poškození firemního majetku.
8. Získání náhrad a příplatků, které mohou být uděleny například za svátky, přesčasy nebo služební cesty.
9. Procházení dat z předchozího bodu a rozdělování hodnot na příplatky za práci ve svátek, takzvané ostatní náhrady a nezdánitelné náhrady.
10. Výpočet náhrad za placené volno podle hodnoty průměrné hodinové mzdy (v případě měsíční mzdy), nebo základní hodinové mzdy a zároveň je také pro hodinový druh mzdy vypočtena základní částka mzdy za daný měsíc podle počtu odpracovaných hodin a počtu svátků v tomto měsíci.
11. Výpočet hrubé mzdy a její zaokrouhlení nahoru na celé Kč.

$$\begin{aligned} \text{Hrubá mzda} &= \text{Základní mzda} + \text{Osobní ohodnocení} + \text{Prémie} \\ &+ \text{Příplatky za svátky} + \text{Náhrady za placené volno} \\ &+ \text{Náhrady za dovolenou} \end{aligned} \quad (2)$$

12. Výpočet zdravotního a sociálního pojištění odváděného zaměstnavatelem za zaměstnance a zaokrouhlení těchto částek nahoru na celé Kč.

$$\text{Sociální pojištění (zaměstnavatel)} = \text{Hrubá mzda} \cdot 0,25 \quad (3)$$

$$\text{Zdravotní pojištění (zaměstnavatel)} = \text{Hrubá mzda} \cdot 0,09 \quad (4)$$

13. Výpočet superhrubé mzdy z hodnot získaných v bodech 11) a 12).

$$\text{Superhrubá mzda} = \text{Hrubá mzda} + \text{Sociální pojištění} + \text{Zdravotní pojištění} \quad (5)$$

14. Výpočet zálohy na daň – tedy daně z příjmu před započtením daňových slev a zvýhodnění. Tato hodnota je 15% superhrubé mzdy zaokrouhlené nahoru na celé storkoruny, na což se ukázal být nejvhodnějším následující postup.

$$\text{\$in["taxbeforediscount"]} = \text{ceil}(\text{\$in["supergross"]} / 100) * 100 * 0.15;$$

Výpis 4: Výpočet zálohy na daň

15. Nyní jsou rozděleny daňové slevy a zvýhodnění, které se započítávají měsíčně, na kategorie sleva na poplatníka, sleva z důvodu zdravotního znevýhodnění (ZP), zvýhodnění na dítě a jiné slevy. Daňová zvýhodnění se od daňových slev liší tím, že zatímco součet slev nemůže snížit zálohu na daň z příjmu „pod“ hodnotu 0 Kč, u daňových zvýhodnění je to možné a zaměstnanci tudíž poté není daň při výpočtu čisté mzdy odečtena, ale je mu přičtena absolutní hodnota částky, o kterou započítání daňového zvýhodnění přesáhlo do záporných hodnot. Tato částka se nazývá daňový bonus.
16. Výpočet zdravotního a sociálního pojištění odváděného zaměstnancem a zaokrouhlení výsledných hodnot nahoru na celé Kč. Zde je potřeba brát v úvahu i fakt, zda je zaměstnanec ve fázi II. pilíře důchodového spoření, jelikož tato skutečnost ovlivňuje výši částky sociálního pojištění. Běžná hodnota sociálního pojištění odváděného zaměstnancem je 6,5% hrubé mzdy, avšak v případě přítomnosti v II. pilíři je tato částka snížena na 3,5% a zbylá 3% jsou započítána do důchodového spoření zároveň s dalšími 2% hrubé mzdy.

$$\text{Sociální pojištění (zaměstnanec)} = \text{Hrubá mzda} \cdot 0,045 \quad (6)$$

$$\text{Zdravotní pojištění (zaměstnanec)} = \text{Hrubá mzda} \cdot 0,065 \quad (7)$$

$$\text{Důchodové spoření} = \text{Hrubá mzda} \cdot 0,05 \quad (8)$$

17. Čistá mzda je vypočítána jako rozdíl hrubé mzdy a pojištění placených zaměstnancem, daně a případně důchodového spoření.

$$\begin{aligned} \text{Čistá mzda} = & \text{Hrubá mzda} - \text{Zdravotní pojištění} - \text{Sociální pojištění} \\ & - \text{Důchodové spoření} - \text{Daň} \end{aligned} \quad (9)$$

18. Sečtení srážek účinných v daném měsíci.
19. Výpočet konečné částky, která je vyplacena zaměstnanci za zvolený měsíc převodem na účet, či v hotovosti. Tato hodnota je pojmenována „Vyúčtování“.

$$\begin{aligned} \text{Vyúčtování} = & \text{Čistá mzda} - \text{Srážky} - \text{Záloha} + \text{Náhrady} \\ & + \text{Nezdanitelné náhrady} + \text{Daňový bonus} \end{aligned} \quad (10)$$

20. Výpočet takzvaných osobních nákladů – to znamená, jakou částku zaměstnavatel za daný měsíc musí za daného zaměstnance vydat na mzdě, pojištění a dani.

$$\begin{aligned} \text{Osobní náklady} = & \text{Superhrubá mzda} - \text{Srážky} + \text{Náhrady} \\ & + \text{Nezdanitelné náhrady} + \text{Daňový bonus} \end{aligned} \quad (11)$$

21. Sestavení SQL dotazu a vložení dat do databáze.

4.4.3 Výstupy

Ze získaných hodnot je za pomoci třídy mPDF, kterou jsem zmiňoval v kapitole 3.2, vygenerováno několik dokumentů ve formátu PDF. Prvními výstupy jsou výplatní pásky a výplatní listy. Výplatní pásky (viz příloha D) jsou rozdávány zaměstnancům, aby měli přibližný přehled, jak bylo dosaženo výsledné částky, která je jejich mzdou. Výplatní listy se spolu se mzdovými listy, přehledy mezd a přehledy o výši pojistného zakládají do archívu, aby bylo možno je předložit k případné kontrole. Kromě těchto dokumentů je ještě možno vytvořit soupis sociálního pojištění a soupisy zdravotního pojištění, které se posílají příslušným úřadům.

5 Užití dovedností v praxi

Přestože hlavní náplní praxe bylo programování v jazyce PHP, jehož výuka není na VŠB příliš podporována, v jiných směrech jsem ocenil dovednosti nabyté během studia.

Při práci s databází jsem využil své znalosti získané během tří semestrů, kdy jsem absolvoval databázové předměty, o tvorbě tabulek, skládání složitějších dotazů nebo práci se systémovým katalogem. Práce s databází MySQL se sice v některých věcech odlišovala od databází SQLServer a Oracle, na jejichž výuku je více soustředěna pozornost, nicméně díky dobrým základům nebylo velkým problémem pochopit, jakým způsobem je například dotaz na databázi pozměnit, aby odpovídal požadavkům MySQL.

Při tvorbě šablon a formulářů mi velmi pomohly znalosti HTML5, které mi velmi usnadnily další zpracovávání získaných dat. Tyto znalosti jsem spolu se základy JavaScriptu získával především během zimního semestru v předmětu o vývoji internetových aplikací a na praxi je dobře zužitkoval.

Co se týče PHP, jak jsem již zmínil, musel jsem se spolehnout spíše na své vlastní znalosti a dokumentaci, avšak hojně jsem využil informací o návrhovém vzoru Transakční skript, jelikož realizace jeho základní myšlenky - zpracování jedné uživatelské akce pomocí jedné funkce - mi při přidávání, úpravě či odebírání záznamů a jejich filtrování velmi vyhovovalo.

6 Chybějící dovednosti

Nejzásadnějším nedostatkem v mých znalostech byla při nástupu na praxi neznalost transformací XSL, jelikož pomocí nich se tvoří ve firmě BINARGON s.r.o. všechny uživatelské výstupy. S touto technologií naštěstí nebylo těžké naučit se pracovat a zároveň s ní jsem se rovněž musel seznámit s jazykem XPath, jehož znalosti jsem poté využil i při studiu.

Další novinkou pro mě byla technologie LESS, která je rozšířením CSS a umožňuje například psaní přehlednějšího CSS kódu díky zanořování tříd nebo využití proměnných. Naučit se psát kód ve stylu LESS spíše než klasického CSS bylo spíše o tom zvyknout si než učit se něco nového.

Kromě těchto věcí jsem se musel naučit práci s API, se kterými jsem nikdy dříve nepracoval, což je ovšem také určitě důležitou zkušeností, kterou ještě v budoucnu využiji.

Co se dalo předpokládat, byla má neznalost mzdového účetnictví při vývoji modulu s touto funkcí. Abych pochopil, jakým způsobem probíhají výpočty jednotlivých složek mzdy, daní nebo pojištění, musel jsem toto téma konzultovat s paní účetní.

Mimo výše zmíněných záležitostí jsem většinu používaných technologií alespoň základně znal a především jsem se tedy zlepšoval v jejich používání. Jako příklad zde mohu uvést vytváření a zpracování HTTP požadavků, ať už v souvislosti s API či formuláři, které jsem prováděl snad každý den praxe, tudíž jsem si psaní zdrojových kódů na toto téma velmi dobře osvojil.

7 Shrnutí výsledků

Co se týče vývoje šablony, ten především sloužil k naučení technologií XSLT a LESS a stejnou šablonu paralelně se mnou zpracovával i zkušenější kolega, jehož verze poté byla nasazena na daný web, nicméně účel, se kterým jsem tvořil svou verzi, byl také víceméně splněn, tudíž lze tento projekt pravděpodobně označit za úspěšný.

Tvorba modulů pro Webmaster Tools a Google Analytics z mého pohledu proběhla úspěšně, nicméně prozatím nemohly být nasazeny do provozu, jelikož se naskytly problémy s ověřováním uživatele pomocí údajů přiřazených jeho Google účtu. Řešení těchto potíží je však spíše pozastaveno díky tomu, že jiné úkoly mají vyšší prioritu, než že by bylo obzvlášť složité.

Modul pro účetnictví bude mít kromě zde zmiňované personalistiky a mzdového účetnictví ještě další složky - příkladem může být třeba majetkové účetnictví - tudíž vývoj v rámci tohoto projektu ještě zdaleka není ukončen. Po dokončení bude navíc potřeba vše důkladně otestovat a výsledky porovnávat s výstupy jiných účetních programů, aby bylo při uvedení do ostrého provozu jisté, že systém funguje, jak bylo zamýšleno.

8 Závěr

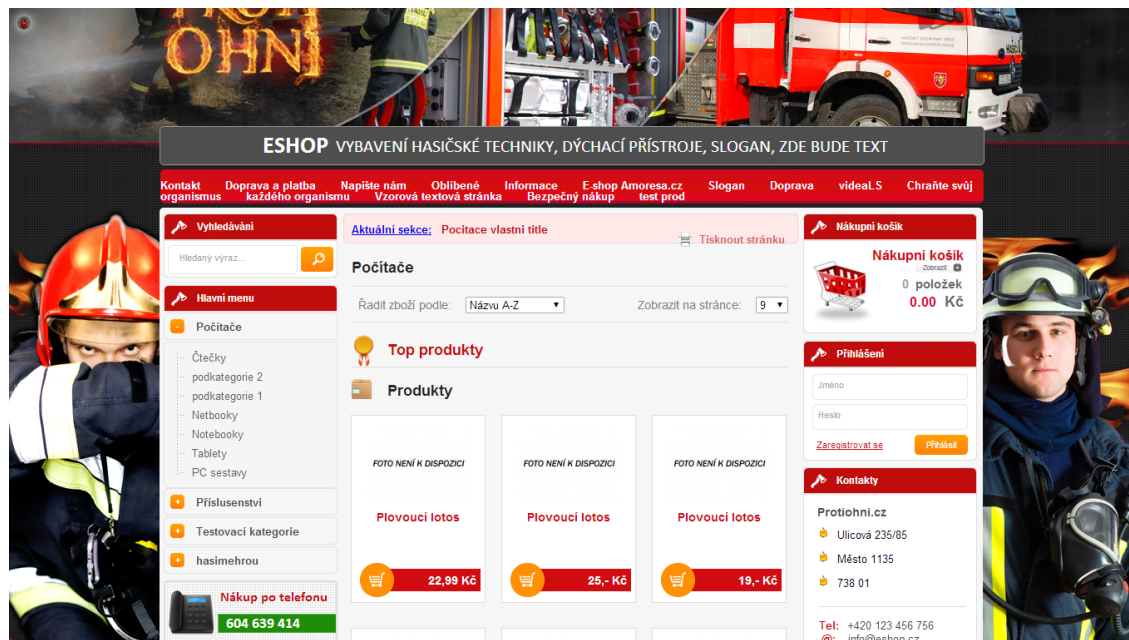
Během praxe jsem pracoval s technologiemi, které jsem dříve neznal, a značně jsem se zdokonalil ve v oblastech, které pro mě nebyly úplně novými. Kromě toho jsem se dozvěděl o poměrech ve vývojářské firmě – například o jednání se zákazníky. Ačkoliv to nebylo přímým účelem, rozšířil jsem své znalosti také v oblasti účetnictví a možností e-shopů.

Ve srovnání s bakalářskou prací na zadané téma je dle mého názoru odborná praxe užitečnější například z toho pohledu, že se student lépe naučí pracovat s kolegy a zároveň tím získá cenný zápis do sekce týkající se praxe na životopisu při hledání zaměstnání.

9 Reference

- [1] BINARGON s.r.o. *Služby pro majitele www stránek a eshopů* [online]. [cit. 2014-04-28]. Dostupné z: <http://www.binargon.cz/sluzby/>
- [2] Usage statistics and market share of PHP for websites. W3TECHS. [online]. 2014-04-28 [cit. 2014-04-28]. Dostupné z: <http://w3techs.com/technologies/details/pl-php/all/all>
- [3] *mPDF Manual* [online]. 2005, 2013-12-31 [cit. 2014-04-28]. Dostupné z: <http://mpdf1.com/manual/>
- [4] Developer's Guide: Protocol - Webmaster Tools. [online]. 2012-03-30 [cit. 2014-04-28]. Dostupné z: https://developers.google.com/webmaster-tools/docs/2.0/developers_guide_protocol
- [5] Dimensions & Metrics Reference - Google Analytics. [online]. 2014-04-28 [cit. 2014-04-28]. Dostupné z: <https://developers.google.com/analytics/devguides/reporting/core/dimsmets>
- [6] *Google Charts* [online]. 2012-04-03 [cit. 2014-04-28]. Dostupné z: <https://developers.google.com/chart/>
- [7] ZDRAŽIL, Vladimír. Jak dlouho archivovat firemní dokumenty? Až 30 let. 2009. Dostupné z: <http://euro.e15.cz/profit/jak-dlouho-archivovat-firemni-dokumenty-az-30-let-895362>

A Šablona e-shopu



Obrázek 2: Šablona e-shopu proti-ohni.cz – úvodní strana

B Webmaster Tools

Úvodní strana Zboží Zakázky Dávkový import Export/Import Komunikace Nastavení Číselníky Plugins System

Plugins » Webmaster tools

podpora@binargon.com
+420 724 844 329
9:00 - 16:00

Odhlásit
uživatele:
binargon

CZ pl en de ru sk hu

Webmaster Tools

http://www.berylo.cz/

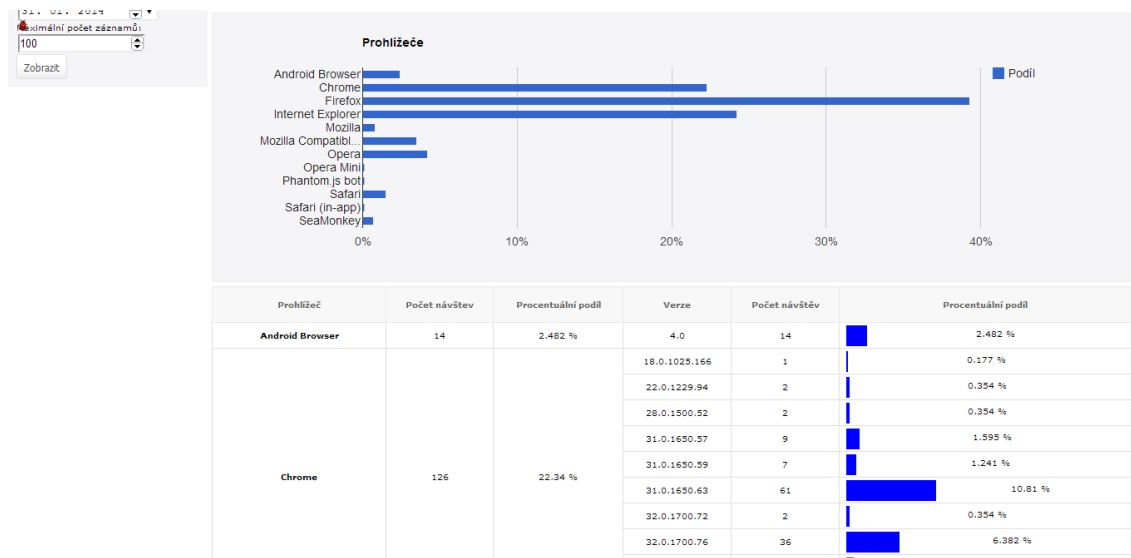
(Ověřeno)

Adresa Obvyčejná WEB Přidat sitemap

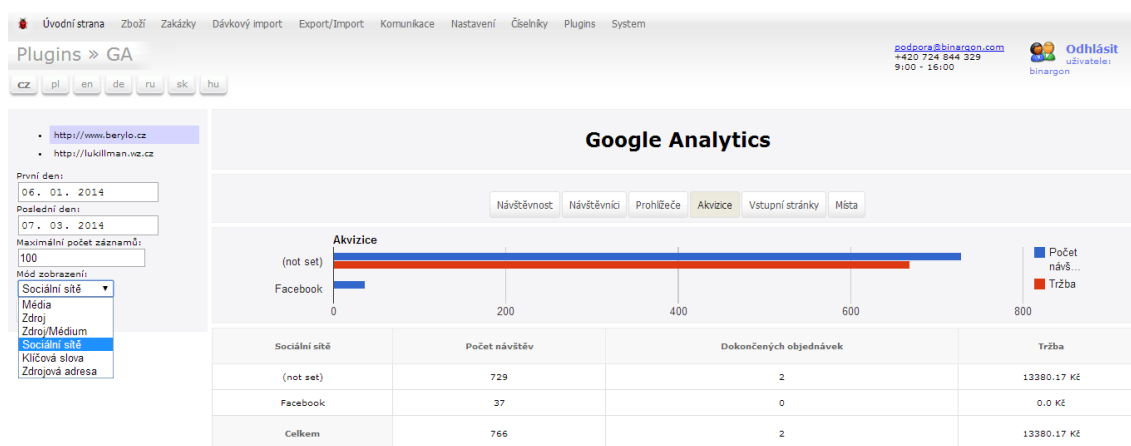
Klíčová slova		Chyby při procházení			
Typ	Slovo	Typ	Čas	Adresa	Detail
external	www.berylo.cz	in-sitemap	2014-01-14T21:26:34.000	http://www.berylo.cz/agility-prekazky/zavodni/	404 (Not found)
external	agility	Odkazován z:	http://www.berylo.cz/o-nas/		
external	agility překážky	Odkazován z:	http://www.berylo.cz/obchodni-podminky/		
external	doprava	Odkazován z:	http://www.berylo.cz/sitemaps_google_1_1.xml		
external	kontakty	Odkazován z:	http://www.berylo.cz/agility-prekazka/		
external	o nás	Odkazován z:	http://www.berylo.cz/		
external	obchodní podmínky	Odkazován z:	http://www.cz.rostat.org/analysis.berylo.cz		
external	partneři	not-found	2014-01-15T03:17:48.000	http://www.berylo.cz/agility-tunel/	404 (Not found)
external	sitemap	Odkazován z:	http://www.berylo.cz/		
external	home page	Odkazován z:	http://www.berylo.cz/sitemaps_google_1.xml		

Obrázek 3: Webmaster Tools - Detail webu

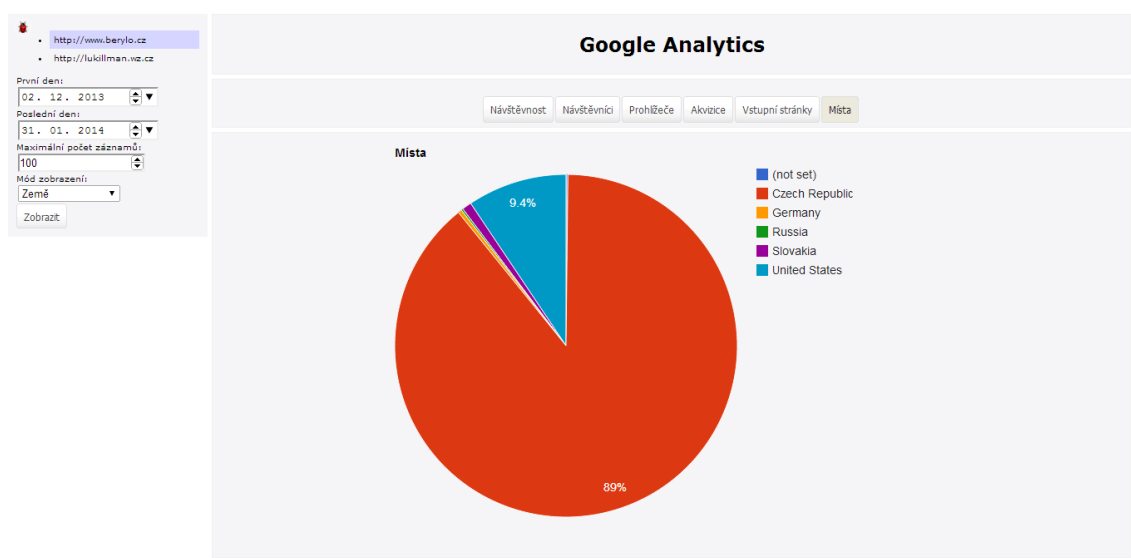
C Google Analytics



Obrázek 4: Google Analytics – Prohlížeče



Obrázek 5: Google Analytics – Akvizice



Obrázek 6: Google Analytics – Místa

D Mzdové účetnictví

Uúčtovné sporení	U	Sociální zaměstnavatel	3414	Zdravotní zaměstnavatel	1430
Sleva na poplatníka	2070	Sleva na invaliditu	0	Ostatní slevy	0
Daň, zvýhodnění na dítě	1117	Daňový bonus	442	Sleva na dítě	675
Nadlimitní příjem	0	Superhrubá mzda	18300	Záloha na daň	0
Čistá mzda	12153	Srážky	123	Náhrady	946
Vyúčtování	14874	Záloha	0	Nezdanitelné náhrady	1456

Majitel webux					
Jméno	Použitý průměr	60.00	Období	1/2014	
Rodné číslo	Zbývajících dovolená	20	Kalendářní dny	31	Prac. dny 22
Nemoc/Ošetřovné	0/0	Neomluveno	0	Odpracováno	9 Hodiny 72
Osobní náklady	5789	Dovolená	0 d	0	Osobní ohodnocení 0
Základní mzda	4320	Svátky	1 d	0	Prémie 0 % 0
Neplacené volno	0 d	Placené volno	0 d	0	Ost. náhrady 0
Hrubá mzda	4320	Sociální zaměstnanec	281	Zdravotní zaměstnanec	195
Důchodové spoření	0	Sociální zaměstnavatel	1080	Zdravotní zaměstnavatel	389
Sleva na poplatníka	2070	Sleva na invaliditu	0	Ostatní slevy	335
Nadlimitní příjem	0	Superhrubá mzda	5789	Záloha na daň	0
Čistá mzda	3844	Srážky	0	Náhrady	0
Vyúčtování	3844	Záloha	0	Nezdanitelné náhrady	0

Majitel webux					
Jméno	Použitý průměr	81.52	Období	1/2014	
Rodné číslo	Zbývajících dovolená	19	Kalendářní dny	31	Prac. dny 22
Nemoc/Ošetřovné	0/6	Neomluveno	0	Odpracováno	17 Hodiny 136
Osobní náklady	18207	Dovolená	0 d	0	Osobní ohodnocení 500
Základní mzda	11086	Svátky	1 d	0	Prémie 18 % 2000
Neplacené volno	6 d	Placené volno	0 d	0	Ost. náhrady 0
Hrubá mzda	13587	Sociální zaměstnanec	884	Zdravotní zaměstnanec	612
Důchodové spoření	0	Sociální zaměstnavatel	3397	Zdravotní zaměstnavatel	1223
Sleva na poplatníka	2070	Sleva na invaliditu	0	Ostatní slevy	0
Nadlimitní příjem	0	Superhrubá mzda	18207	Záloha na daň	675
Čistá mzda	11416	Srážky	0	Náhrady	0
Vyúčtování	11416	Záloha	0	Nezdanitelné náhrady	0

Majitel webux					
Jméno	Wells Herbert George	Použitý průměr	87.40	Období	1/2014
Rodné číslo	6620311734	Zbývajících dovolená	0	Kalendářní dny	31 Prac. dny 22

Obrázek 7: PDF s výplatními páskami